

REMARKS

I. Introduction

In response to the Office Action dated November 21, 2003, claims 1, 3, 5, 13, 15, 17, 20, 21, and 25 have been amended. Claims 1-36 remain in the application. Re-examination and re-consideration of the application, as amended, is requested.

II. Drawings

Pursuant to the object to FIG. 4, Applicants have reoriented certain text in FIG. 4. Accordingly, Applicants request withdrawal of the objection to the drawings.

III. Office Action Subject Matter Rejection

On page (3), the Office Action rejects claim 1 under 35 U.S.C. § 101 as being directed to non-statutory subject matter. The Applicants have amended claim 1 to provide for stylizing data and providing the stylized data for viewing. Accordingly, Applicants believe that the claims currently describe statutory subject matter and request withdrawal of the rejection.

Should issues still remain in this regard, the Applicants requests that the Examiner indicate how the rejection can be overcome and how problems may be resolved, in accordance with the directives of the Examination Guidelines for Computer-Related Inventions. See Guidelines II M.P.E.P. § 2106. Specifically, should it be necessary, the Applicants request that the Examiner identify features of the invention that would render the claimed subject matter statutory if recited in the claim. See Guidelines IV, M.P.E.P. § 2106.

IV. Non-Art Rejections

In paragraph (4) of the Office Action, claims 1-12 were rejected under 35 U.S.C. §112, first paragraph, as failing to comply with the enablement requirement. Applicants submit that under 35 U.S.C. §112, first paragraph, the specification must be enabling and not the claims. In this regard, the specification must contain a written description of the invention that enables any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the invention.

Examining the specification, the step of obtaining a domain object is clearly supported on page 17, lines 21-23. The step of obtaining an application object is clearly supported on page 17, line 24-page 18, line 13. The amended step of stylizing the presentation object is clearly supported on page 18, lines 14 – page 19, line 8.

Nonetheless, to more clearly depict the invention, Applicants have amended claim 1 as indicated above.

In view of the above, Applicants respectfully request withdrawal of the above rejection.

V. Prior Art Rejections

In paragraph (8) of the Office Action, claims 1-36 were rejected under 35 U.S.C. §102(e) as being anticipated by Helgeson et al., U.S. Patent No. 6,643,652 (Helgeson).

Applicants respectfully traverse this rejection.

Specifically, claim 13 was rejected as follows:

As to claim 13, Helgeson et al. (hereinafter referred as Helgeson) discloses a system as claimed by applicant for stylizing data (or transform data) [e.g., see Abstract, lines 3-9; col. 2, lines 51-67]) in a network [e.g. see, Fig. 1], comprising:

- a) an objected-oriented computer system having memory and a data storage device coupled thereto [e.g., see 211, 209, 217, 219, 221, 223, Fig. 2; col. 5, lines 13-14];
- b) a domain object stored in the memory of the computer, the domain object comprising an object representation of data stored in a database for a domain entity [e.g. the business object of the fgt-dd-class which stored in the meta-data store as a database table as shown at col. 14 , line 23-col. 15, line 53];
- c) an application object stored in the memory of the computer, the application object comprising an object representation of the data in the domain object that is relevant for a particular computer application [the Applications (507), Fig. 5];
- d) a presentation object stored in the memory of the computer, the presentation object comprising an object representation that encapsulates a visual appearance of the data in the application object [e.g. the XSLT stylesheet, col. 51, lines 31-34].

Applicants traverse the above rejections for one or more of the following reasons:

- (1) Helgeson fails to teach, disclose or suggest three discrete objects – a domain object, an application object, and a presentation object that are based on each other as claimed;
- (2) Helgeson fails to teach, disclose or suggest a portable stylization process that is configured to stylize the domain object into an application object;
- (3) Helgeson fails to teach, disclose or suggest an application object that is an object representation of data in a domain object that is relevant for a particular computer application;

(4) Helgeson fails to teach, disclose or suggest a portable stylization process configured to stylize an application object into a presentation object; and

(5) Helgeson fails to teach, disclose or suggest a presentation object that encapsulates a visual appearance of data in an application object.

Independent claims 1, 13, and 25 are generally directed to the ability stylize data in discrete portable steps. As used in the claims and set forth in the specification, “stylization” refers to the process wherein data is transformed from its pure/raw form to the final presentation desired by an application (see page 3, line 22- page 4, line 2). The portability of the stylization process (as claimed) allows stylization to be spread across multiple computers or tiers in a client-server environment.

The claims initially provide for obtaining a domain object. The domain object provides an object representation of data stored in a database for a domain entity.

A portable stylization process then provides the ability to stylize the data in the domain object into an application object. The application object is an object representation of the data in the domain object that is relevant for a particular computer application.

Another portable stylization process then stylizes the application object into a presentation object. The presentation object is an object representation that encapsulates a visual appearance of the data in the application object.

Thus, with the two portable stylization processes and the three different objects (i.e., the domain object, application object, and presentation object), stylization is broken up into multiple discrete parts that provides considerable flexibility.

The cited reference does not teach nor suggest these various elements of Applicants’ independent claims. Nor does the cited reference provide the flexibility offered by the present invention as claimed.

Helgeson merely describes a mechanism for managing data exchange among systems in a network. The systems and methods of the present mechanism translate data from a system specific local format to a generic interchange format object, and vice versa, with predefined stylesheets using generic components and a system specific service components which utilize a native application programming interface of the specific local system (see Abstract).

The Office Action relies on the Applications 507 depicted in FIG. 5 to teach the application object as claimed. However, as noted above, the claimed application object is an object

representation of data in the domain object that is relevant for a particular computer application. In addition, the application object is created or stylized by a portable stylization process. However, as indicated in col. 6, lines 4-10, application layer 507 merely “provides objects and services particular to a given application.” There is no provision in Helgeson that indicates that there are three separable objects as claimed (i.e., a domain object, application object, and presentation object). Further, Helgeson fails to indicate that a portable stylization process created the application object (as claimed). Instead, Helgeson merely indicates that an application layer provides application specific objects and services.

The Office Action also relies on XSLT stylesheets in col. 51, lines 31-34 to teach the presentation object. However, contrary to that stated in the Office Action, an XSLT stylesheet is not an object. Instead, an XSLT stylesheet is an extensible stylesheet language transformation (see col. 50, lines 24-25). As defined at <http://www.techweb.com/encyclopedia/defineterm?term=xslt> :

XSLT is the processing component of the Extensible Stylesheet Language (XSL). XSLT is widely used to convert XML to HTML for screen display, but can be used to convert to PDF, another XML document or any other format. The conversion is accomplished with an XSLT processor, which transforms the input based on XSLT extensions of the XSL style sheet. XSL statements are also followed. The processor requires an XML parser to separate the XML elements into a tree structure which the processor manipulates. Xpath is a component of XSL that is used for identifying input, calculating numbers and manipulating characters. See XSL, DOM and SAX.

Such language clearly indicates that a stylesheet is not an object as claimed. Further, the output produced is an XML document or a document in another format and is not an object representation that encapsulates a visual appearance of the data in the application object. In this regard, not only is the XSLT not an object, but the XSLT is not based on an application object as claimed. Further, Helgeson also fails to indicate or use a portable stylization process that stylizes the application object into the presentation object.

Moreover, the various elements of Applicants’ claimed invention together provide operational advantages over Helgeson. In addition, Applicants’ invention solves problems not recognized by Helgeson.

Applicants also note that the dependent claims provide further limitations not addressed or suggested by Helgeson. For example, claims 22-24 provide specify the types of domains in which the application may be used. Specifically, the claims provide that the domains may be a mechanical domain, an architecture, engineering, and construction (AEC) domain, or a geographic information

system. These limitations further enhance the independent claims and thereby indicate that the data in the domain objects are relevant in those specific domains. However, claims 22-24 were merely rejected by stating that these domains are a default nature of a domain object in an internet data exchange computer system. Applicants respectfully disagree with and traverse such a statement. Nowhere in Helgeson is there any reference to either a mechanical, AEC, or geographic information environments, implicitly or explicitly. In fact, separate electronic searches of Helgeson for the terms mechanical, AEC, and geographic provide no results whatsoever. Without even mentioning these terms, Helgeson cannot possibly teach these claims. Further, no art was cited that suggests an internet data exchange computer system by default nature provides for such domain fields.

Thus, Applicants submit that independent claims 1, 13, and 25 are allowable over Helgeson. Further, dependent claims 2-12, 14-24, and 26-36 are submitted to be allowable over Helgeson in the same manner, because they are dependent on independent claims 1, 13, and 25, respectively, and thus contain all the limitations of the independent claims. In addition, dependent claims 2-12, 14-24, and 26-36 recite additional novel elements not shown by Helgeson.

VI. Conclusion

In view of the above, it is submitted that this application is now in good order for allowance and such allowance is respectfully solicited. Should the Examiner believe minor matters still remain that can be resolved in a telephone interview, the Examiner is urged to call Applicants' undersigned attorney.

Respectfully submitted,

GATES & COOPER LLP
Attorneys for Applicant(s)

Howard Hughes Center
6701 Center Drive West, Suite 1050
Los Angeles, California 90045
(310) 641-8797

Date: February 20, 2004

By: Jason S. Feldmar
Name: Jason S. Feldmar
Reg. No.: 39,187

JSF/sjm